	Pruning ratio	Oracle estimate	Init. [8, 25]	Pre-def [18]	EPI (ours)
ResNet50	10%	76.90	76.80	76.74	**76.83**$_{(5)}$
	20%	76.37	76.23	76.32	**76.34**$_{(9)}$
	30%	75.78	74.88	75.60	**75.75**$_{(11)}$
	40%	74.93	73.73	74.85	**74.91**$_{(12)}$
	50%	73.89	71.92	73.62	**73.84**$_{(12)}$
	60%	71.97	70.04	**71.92**	71.79$_{(12)}$
	70%	69.00	66.65	68.49	**68.92**$_{(13)}$
	80%	63.77	60.65	62.68	**63.40**$_{(17)}$
	90%	61.65	48.73	**61.65**	**61.65**$_{(30)}$
ResNet34	10%	73.76	73.58	**73.67**	73.59$_{(5)}$
	20%	72.56	72.24	**72.42**	72.31$_{(10)}$
	30%	70.63	**70.53**	70.24	70.28$_{(13)}$
	40%	68.20	**68.20**	68.04	68.00$_{(7)}$
	50%	65.54	65.52	65.22	**65.53**$_{(13)}$
MobileNetV1	10%	72.50	72.34	**72.40**	72.27$_{(5)}$
	20%	71.66	71.48	**71.59**	**71.59**$_{(5)}$
	30%	70.69	70.38	**70.56**	70.49$_{(6)}$
	40%	69.22	**69.09**	69.07	69.05$_{(12)}$
	50%	67.25	67.10	**67.15**	67.03$_{(5)}$
all nets avg acc drop		–	1.378	0.214	**0.142**
automatic		✗	✓	✗	✓

Table 4. The detailed top1 accuracy (in %) of the network with **gradient**-based neuron pruning, under different policies as in Section 4.2 of the main paper. Each reported value under our proposed EPI policy is in the format of [top1 acc]$_{\text{(prune epoch)}}$. The summarize of the accuracy drop is the averaged accuracy drop (in %) compared to the oracle method.

	prune ratio	Oracle estimate	Init. [8, 25]	Pre-def [18]	EPI (ours)
ResNet50	10%	76.9	74.50	**76.90**	76.76$_{(15)}$
	20%	76.25	72.71	**76.13**	76.12$_{(17)}$
	30%	75.56	70.69	75.42	**75.46**$_{(18)}$
	40%	74.53	–	74.46	**74.49**$_{(20)}$
	50%	72.95	–	72.71	**72.90**$_{(26)}$
ResNet34	10%	73.78	73.49	**73.71**	73.64$_{(15)}$
	20%	72.97	72.20	72.78	**72.86**$_{(16)}$
	30%	71.63	71.40	71.53	**71.58**$_{(19)}$
	40%	69.94	67.90	69.78	**69.84**$_{(21)}$
	50%	67.42	–	66.19	**66.97**$_{(24)}$
MobileNetV1	10%	71.87	–	71.65	**71.87**$_{(5)}$
	20%	70.59	–	70.34	**70.58**$_{(9)}$
	30%	68.91	–	**68.88**	68.52$_{(6)}$
	40%	66.60	–	**66.60**	66.50$_{(15)}$
	50%	63.28	–	**63.11**	**63.11**$_{(30)}$
all nets avg acc drop		–	2.023[a]	0.204	**0.132**
automatic		✗	✓	✗	✓

[a] is averaged over the trainable pruning results.

Table 5. The detailed top1 accuracy (in %) of the network with **magnitude**-based neuron pruning, under different policies as in Section 4.2 of the main paper. "–" refers to pruning leads to not-trainable network. Each reported value under our proposed EPI policy is in the format of [top1 acc]$_{\text{(prune epoch)}}$. The summarize of the accuracy drop is the averaged accuracy drop (in %) compared to the oracle method.

Appendices

We provide more experimental details in the following sections.

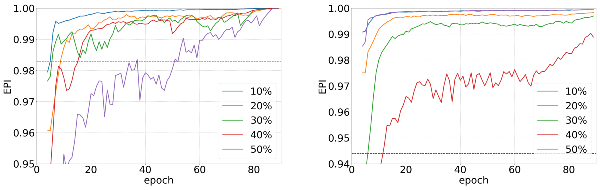
A. Additional EPI-guided pruning results

We perform EPI-guided pruning, with gradient-based criterion, on ResNet50, ResNet34 and MobileNetV1 with different prune ratios and get the final accuracy as shown in Tab. 4. It can be observed that over all pruning ratios, EPI demonstrates superior capability over prior methods in achieving similar performance to oracle while pushing the start of pruning earlier into training.

Similarly, for magnitude-based pruning in Tab. 5 we show the results of pruning with the magnitude universal threshold. In this case, as before, EPI-guided pruning performs better than heuristic pruning. Note that in magnitude-based pruning, pruning at initialization (heuristically pruning at epoch 0) tends to lead to non-trainable networks given a more challenging task amid model compactness.

B. Stability analysis for MobileNetV1

We provide here additional stability analysis results for a MobileNetV1 architecture. As in Section 4.5 of the main paper, the goal is to demonstrate that the sub-network architecture varies significantly during the early stage of training and then slowly converges to the final architecture as the training progresses. In this case, we train a MobileNetV1 to convergence and, in the process, compute the EPI value for different prune ratios. Fig. 7(a) and (b) show the EPI curves with magnitude-based and gradient-based, respectively. Consistent with our main paper results, the stability indicator increases rapidly in the early stage of training and continues increasing steadily for later training stages.



(a) magnitude-based EPI (b) gradient-based EPI

Figure 7. Structure stability analysis for MobileNetV1 for magnitude-based (a) and gradient-based (b) pruning with different ratios. Dashed line in black shows the EPI threshold.

C. Results on MobileNetV3

Fig. 8 shows results for one additional experiment on MobileNetV3 to verify the generalization of the proposed method. Results on this architecture are consistent with the results shown in the paper: the dominant sub-network quickly emerges and tends to be stable earlier for a smaller prune ratio.

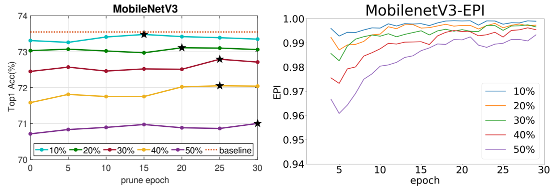


Figure 8. Left: Prune a Mobilenet-V3 with gradient-based method. Right: The EPI value for MobileNetV3 in the first 30 epochs.

D. Comparison to other similarity criteria

In this experiment, we aim at comparing our proposed EPI to other ranking criteria to measure the difference between two network structures. To this end, we consider two ranking correlation measures (spearman and Kendall tau) and the instability measure proposed in [11]. Ranking correlation approaches directly measure the difference in the neuron ranking. These ranking correlation measures require all the neurons in the architecture and can not discriminate between different pruning ratios. Instability focuses on identical architectures trained with different SGD noise and is computed after training is completed. Fig. 9 and Fig. 10 show the results for this experiment on ResNet50. As expected, using other measures, we not only can not distinguish different pruning ratios but also do not have enough discriminative power even during the early stages of training. In contrast, as our approach focuses only on the architecture changes can, as shown in Fig. 6 in the main paper, provide better insights into the stability of the network.

E. EPI-guided Pruning on Object Detection

In this experiment, we extend our proposed EPI policy to object detection. We use a Single Shot multibox Detector (SSD) [26] as our detection network, with a ResNet34 backbone and an input size resolution of 300 × 300. We run the experiments on PASCAL VOC07+12 (union of VOC2007 and VOC2012) [9].

For training, we use PyTorch Distributed Data Parallel and mixed precision. We train the model for 800 epochs in total, with an individual batch per GPU of 128. The learning rate is warmed up linearly to $8e-3$ in the first 50 epochs, remains at the maximum value until epoch 600, and decays every 50 epochs. As an upper bound and baseline, we consider the accuracy of the unpruned model where we obtained 76.8% mAP.

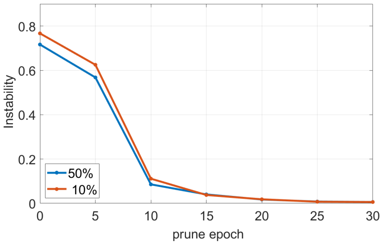


Figure 9. Instability [11] of the networks obtained by pruning at different epochs. In this case, the prune ratio is involved, but the curves are not distinguishable among different prune ratios. Moreover, such instability can only be calculated after training.

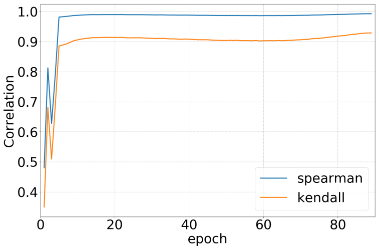
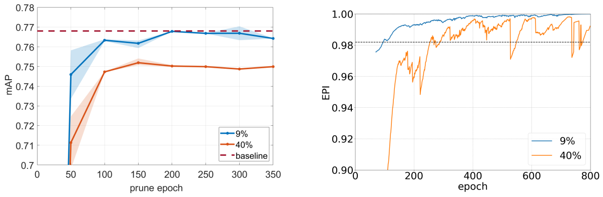


Figure 10. Kendall and Spearman correlation calculated based on gradient-based neuron metric ranking. Each value is the calculated between epoch x and epoch $x-1$. With rank correlation, no prune ratio would be involved. That means if we set a threshold to the correlation value, then for whatever prune ratio, we need to perform pruning at the same epoch.



(a) (b)

Figure 11. Object detection using SSD300-RN34 on Pascal VOC. (a) The magnitude-based pruning result and (b) Structure stability analysis (EPI curve) for SSD300-RN34. Dashed line in black shows the EPI threshold.

We test magnitude-based pruning with 9% and 40% pruning ratio and compare the performance to grid-search pruning with 50 epochs. Fig. 11(a) shows the results for this experiment. As we can see, pruning too early leads to large accuracy drops. However, if we delay the pruning epoch, especially with a lower pruning ratio, we do achieve on-par accuracies with the upper bound.

We also compute the EPI value, see Eq.(6) in the main paper, for these two pruning ratios. In this case, different from our classification set up, we calculate the sub-network structure similarity among the past 50 epochs, *i.e.*, $r = 50$. Fig. 11(b) shows the EPI curve for this experiment. As shown, the EPI value increases rapidly at the early stage of training and then increases gradually as the training progresses. The tendency is consistent with or pruning results in Fig. 11(a). As in our previous experiments, we set an EPI threshold to the magnitude universal threshold value $\tau = 0.983$. Given this threshold, for this architecture, using EPI-guided pruning leads to pruning in the 96th epoch for 9% pruning ratio and pruning in the 255th for a pruning ratio of 40%. The mAp drop with respect to the grid-search result is 0.589% and 0.212% mAP respectively.